#4

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
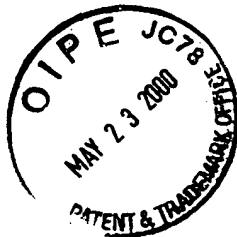
In re Patent Application of

LI et al

Serial No.   09/493,258   Atty Ref.:   723-824

Filed: 28 January 2000   Group: 2772

For: INCREMENTAL INTERLACE   Examiner:
       INTERPOLATION FOR TEXTURE
       MORPHING
                    *   *   *   *   *   *   *   *   *   *   *

                                                   May 23, 2000

Assistant Commissioner for Patents
Washington, DC 20231

SUBMISSION OF FORMAL DRAWINGS

Sir:

Enclosed herewith are eleven sheets of formal, inked drawings for the above-

identified application.

                              Respectfully submitted,

                              NIXON & VANDERHYE P.C.

                    By: _____

                              Robert W. Faris
                              Reg. No. 31,352

RWF:ejs
1100 North Glebe Road, 8th Floor
Arlington, VA 22201-4714
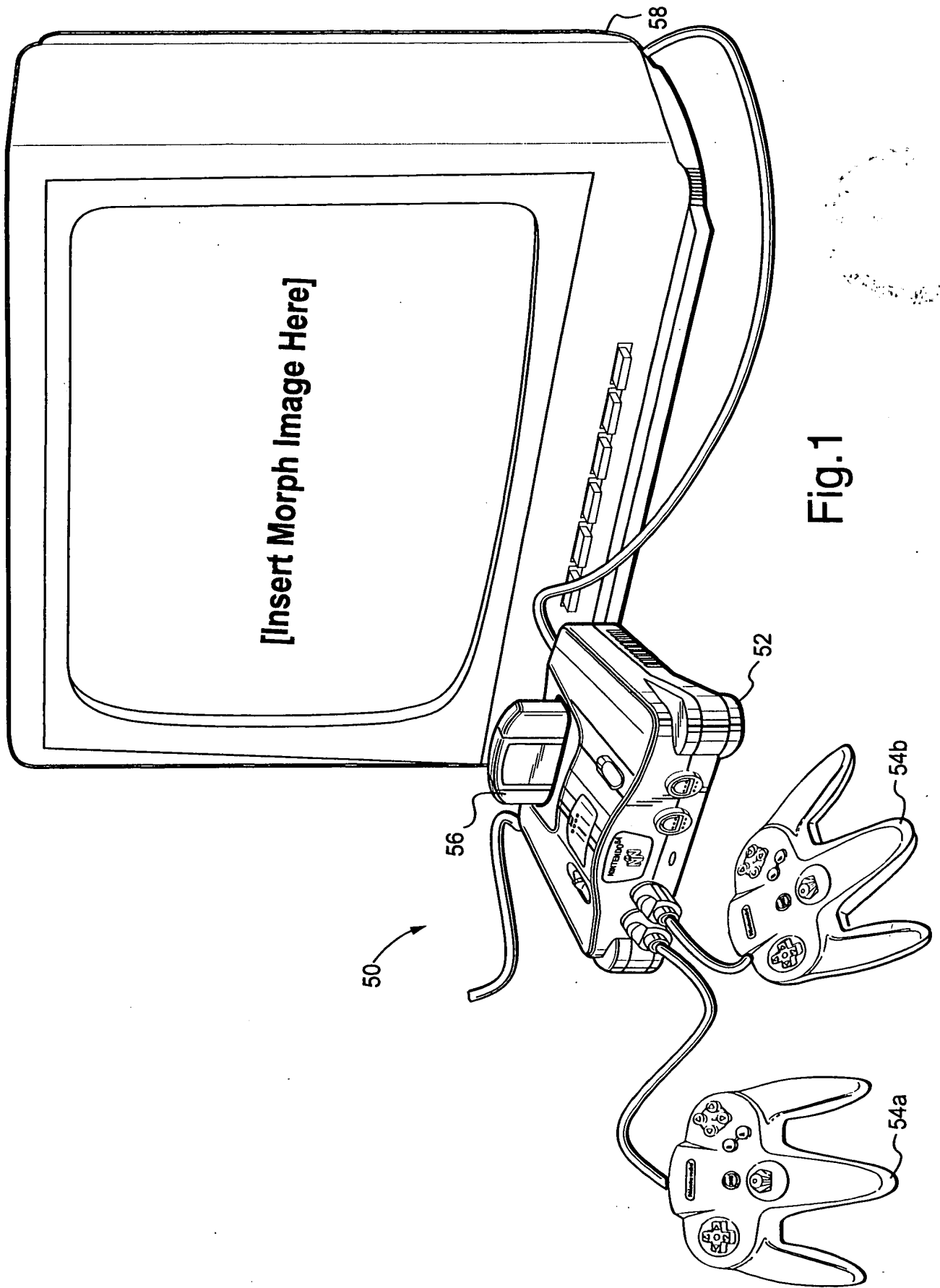Telephone: (703) 816-4000
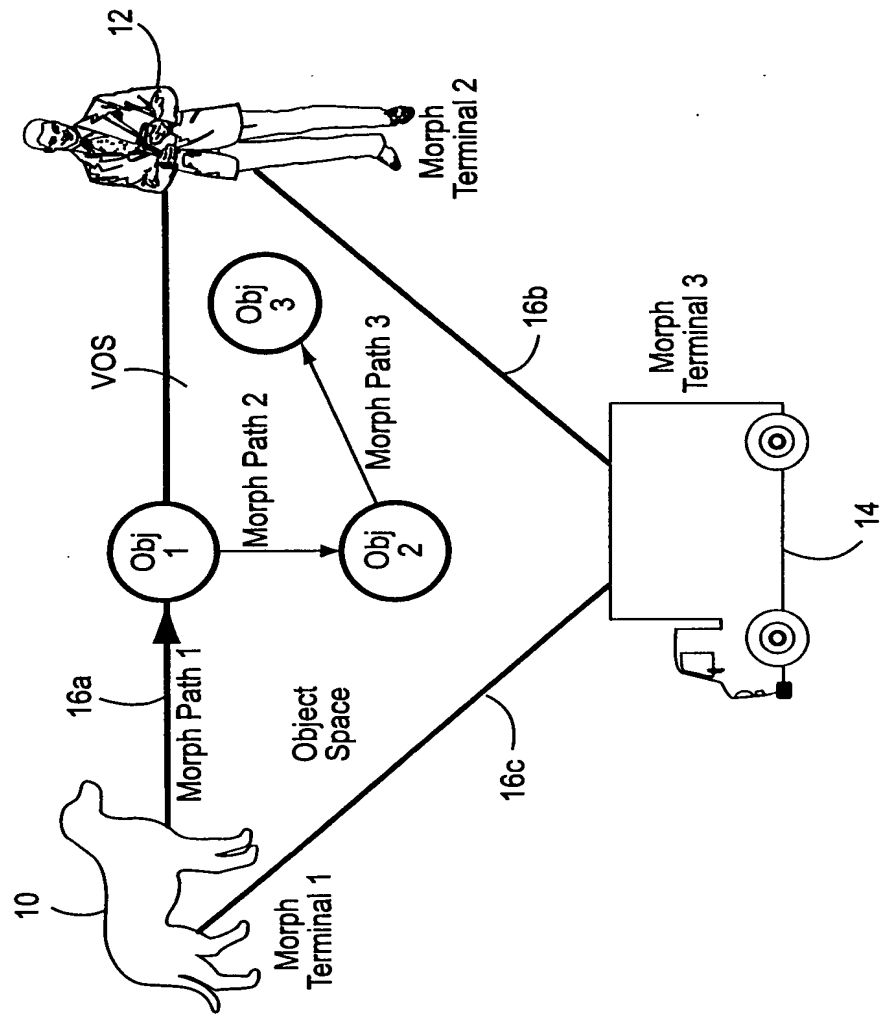Facsimile: (703) 816-4100

[Insert Morph Image Here]
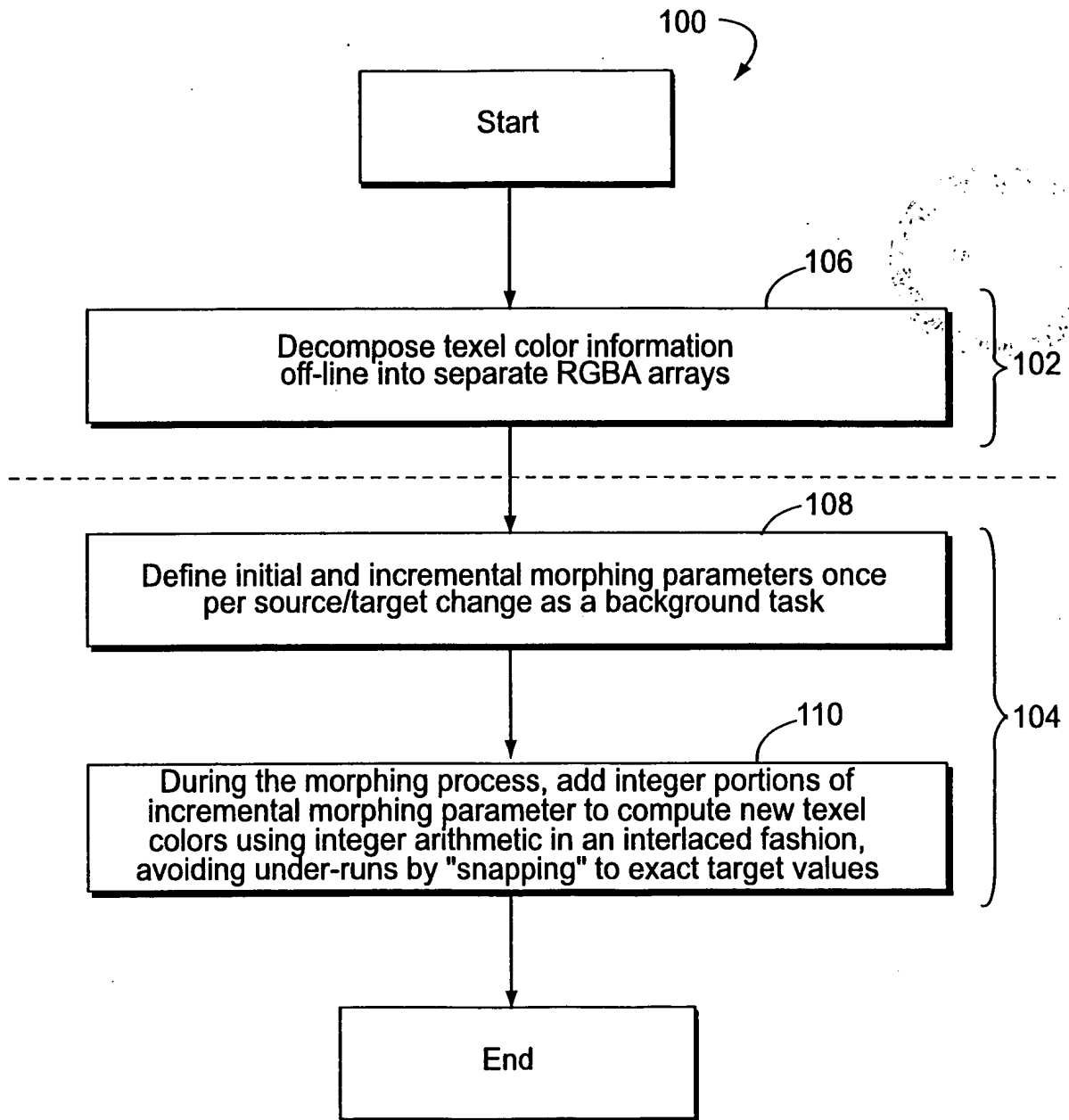
58

52

56

50

54b

54a

Fig.1

Fig. 2

VOS

Morph Path 1

16a

Morph Path 2

Obj 1

Obj 2

Morph Path 3

Obj 3

Object Space

Morph Terminal 1

10

12

Morph Terminal 2

16b

Morph Terminal 3

16c

14

Fig. 3

100

Start

106

Decompose texel color information
off-line into separate RGBA arrays

102

- -

108

Define initial and incremental morphing parameters once
per source/target change as a background task

110

During the morphing process, add integer portions of
incremental morphing parameter to compute new texel
colors using integer arithmetic in an interlaced fashion,
avoiding under-runs by "snapping" to exact target values

104

End

Fig. 3A

56

56a

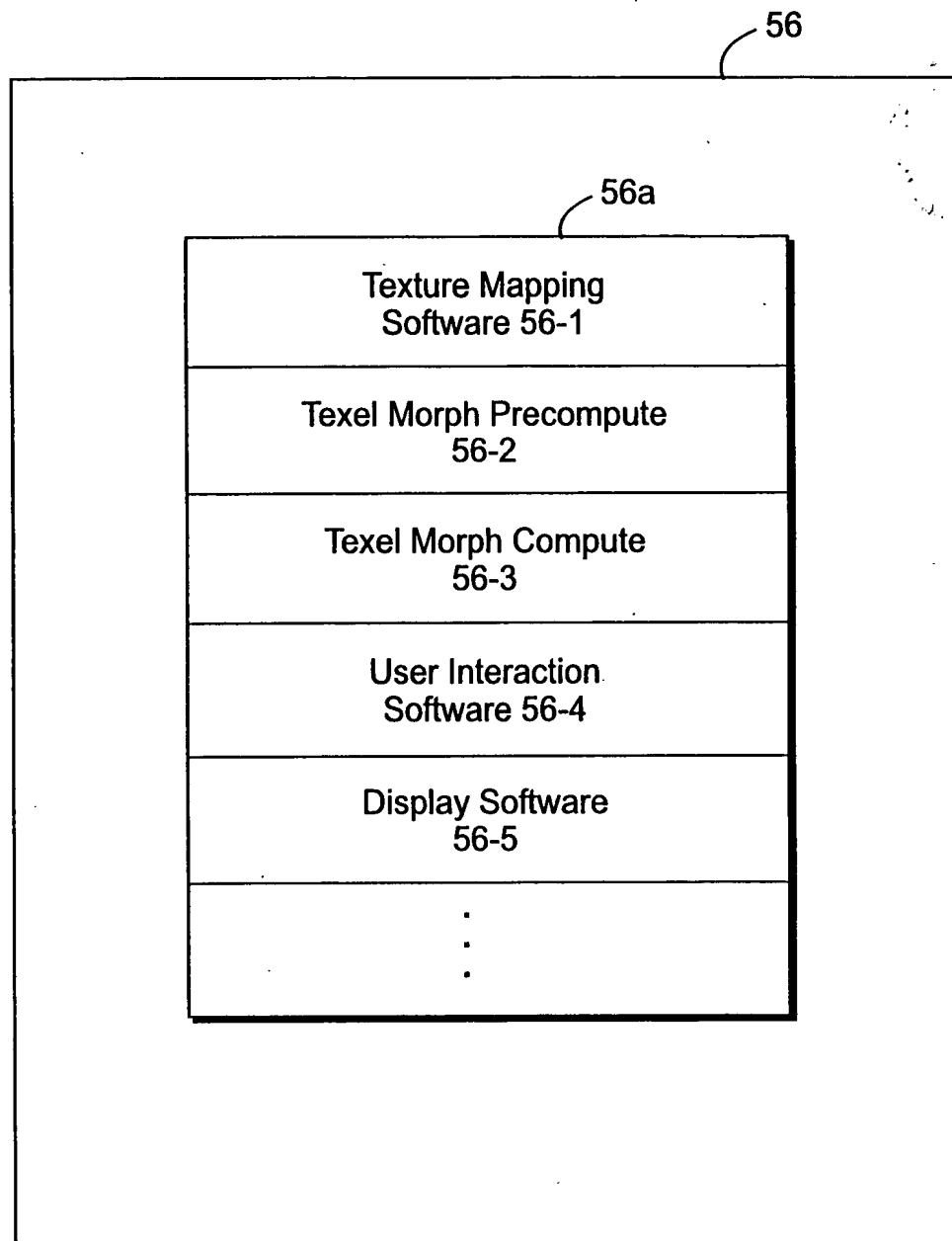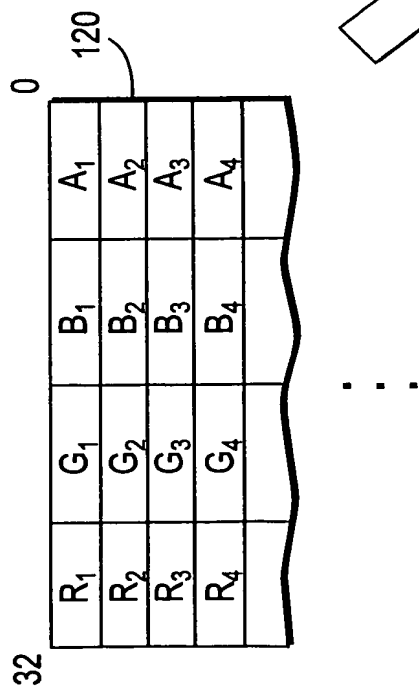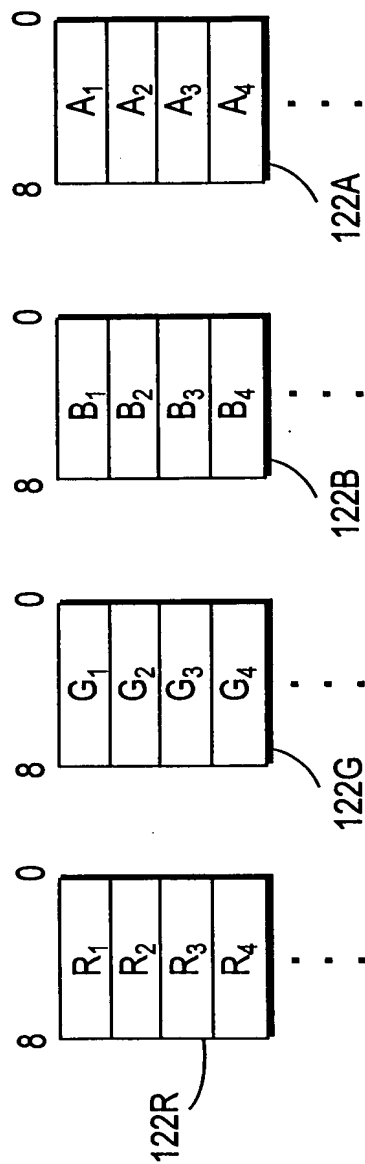| Texture Mapping Software 56-1 |
|---|
| Texel Morph Precompute 56-2 |
| Texel Morph Compute 56-3 |
| User Interaction Software 56-4 |
| Display Software 56-5 |
| ⋮ |

Fig. 4 Offline Color Description
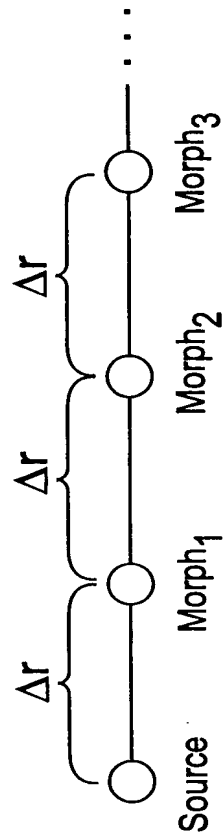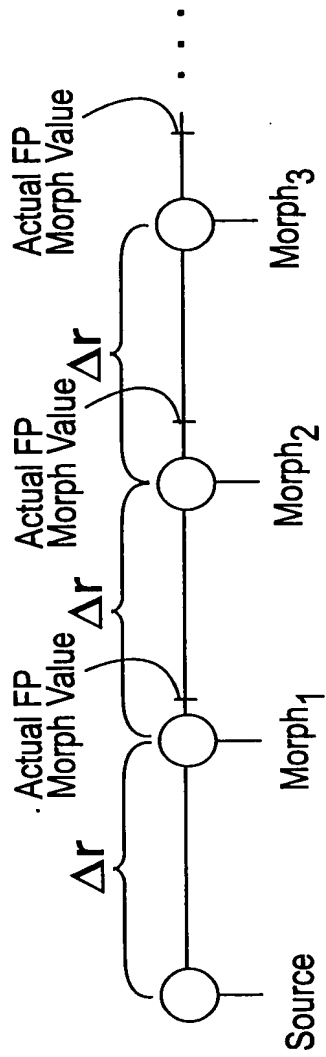
Fig. 5 Uniform Incremental Interpretation

Fig. 5A Error Due to Integer Arithmetic

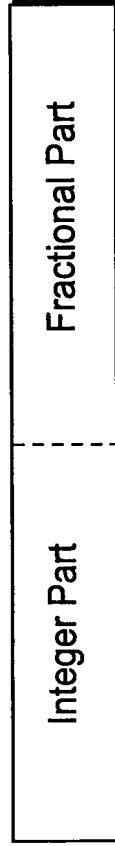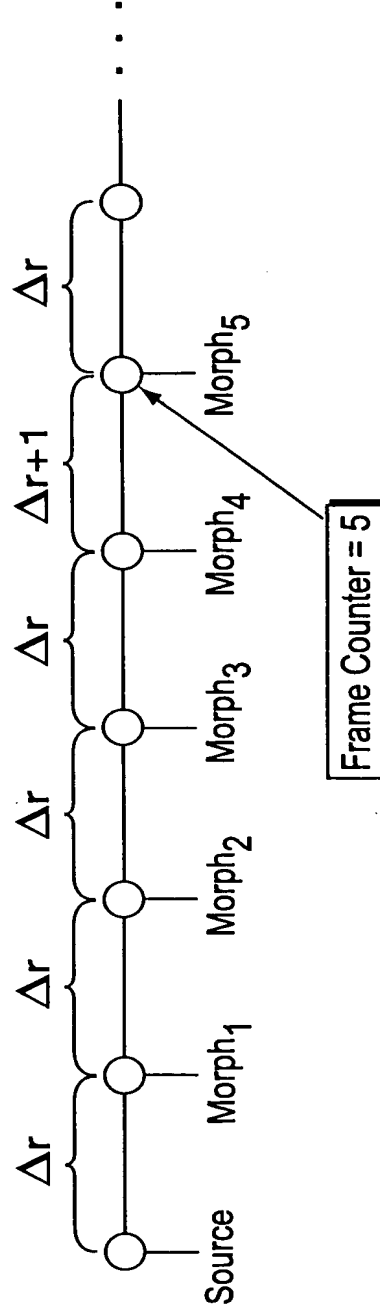| Integer Part	Fractional Part

Fig. 5B Incremental Parameter

Fig. 5C Integer Correction Factor
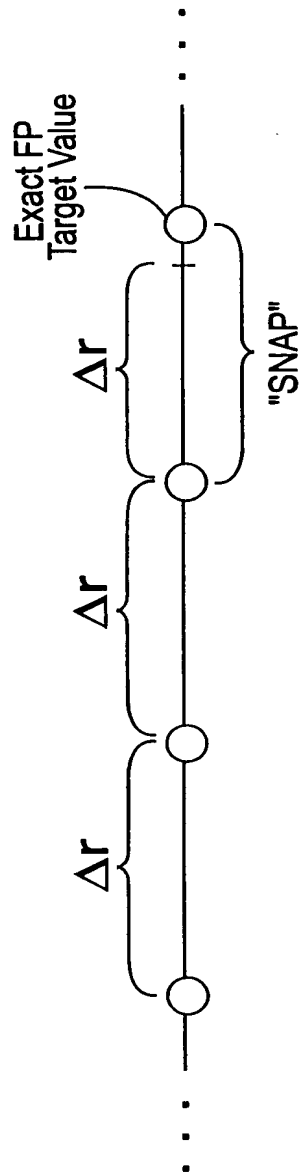
Fig. 5D "Snapping" to Prevent Underrun Autofactor

600a   600c   600b   600a   600c   600b   600a
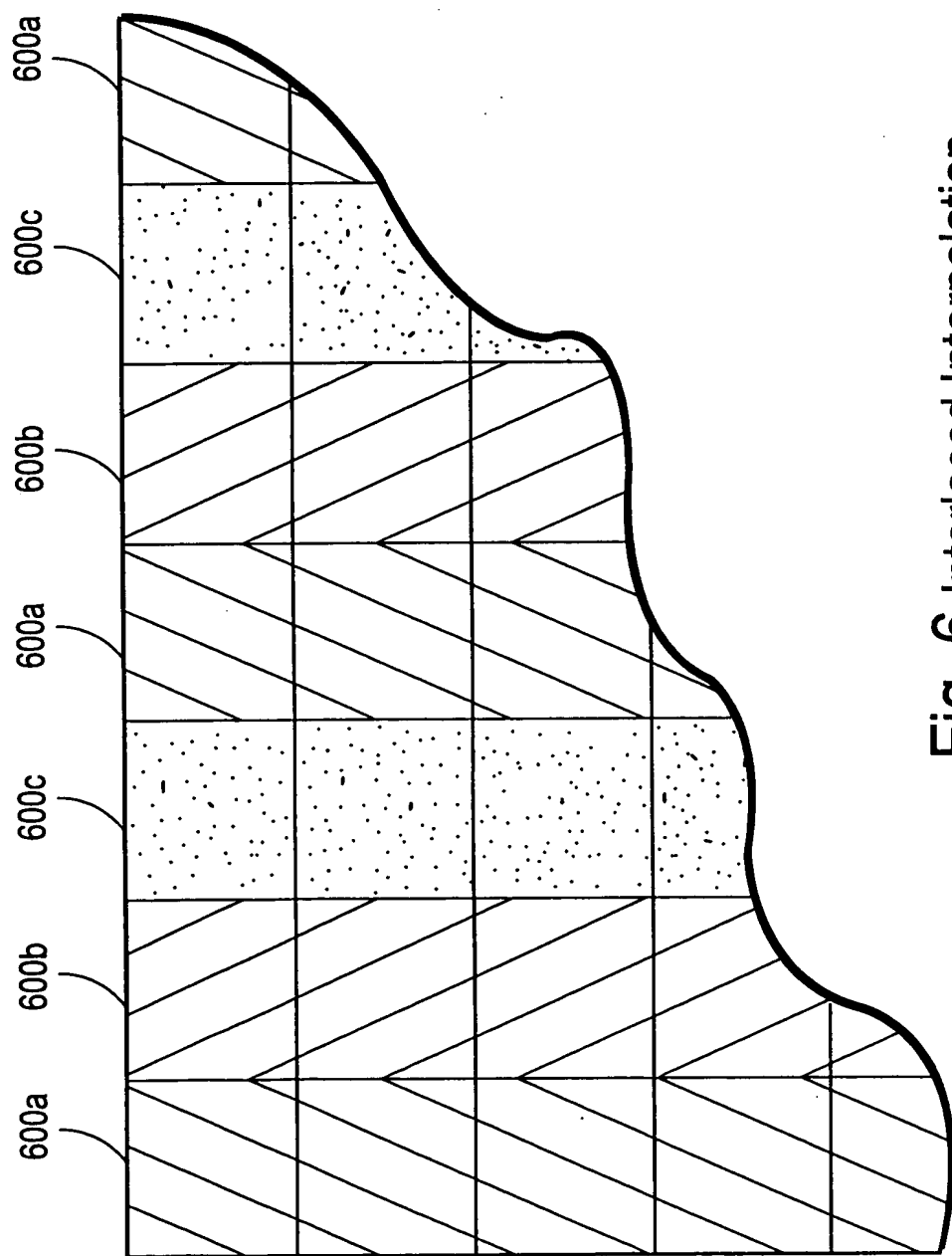
Fig. 6 Interlaced Interpolation

300 ⤵

302 ⤵

<u>Main Thread</u>

306

(Morph
process
starts)

308 ⤵
Sprout a background
thread

304 ⤶

<u>Background Thread</u>

Source
and Target

310 ⤶

For each texel, do:{
  Compute incremental integer;
  Compute frame counter value;
  Multiply incremental value by
      INTERLACE;
  Make sure no "over-runs";
}

312 ⤶
Other computational
duties

314
No ◁—— Parameters
ready

Ready

Yes

316 ⤶
Calculate multiples of
currentFrame

318 ⤶
For each texel, do {
  add incremental value for each component;
  If (currentFrame is a multiple of its frame counter value)
      add extra value specified by constant INTERLACE:
}

320
Last morph
frame? ——Yes——▷

324 ⤶
Snap to target
texture

No

InterlaceCounter++:
If (InterlaceCounter is a multiple of INTERLACE)
  currentFrame++:

322 ⤶

(Morph
process ends)

326 ⤶

Fig. 7 Example Flowchart